**BANGABANDHU SHEIKH MUJIBUR RAHMAN**

**SCIENCE & TECHNOLOGY UNIVERSITY**

GOPALGANJ-8100



***Project Report***

Course Title : Project

Course Code: **CSE-178**

**Project Name: Calendar Application**

**Prepared for:**

#### **F. M. Rahat Hasan Robi**

**Lecturer**

**Department of Computer Science and Engneering**

**BSMRSTU, Gopalganj-8100**

**Prepared by:**

**Md. Sahin Alom**

**ID no: 18CSE051**

**Year :1st; Semester : 2nd**

**Session : 2018-2019**

***Date of Submission*: 30th nov, 2020**

**Faculity of Engneering**

**Department of Computer Science and Engneering**

**BSMRSTU, Gopalganj-8100 ©2019**

This project on **Calendar in C** programming language is a console application without graphics. To make the calendar look colorful, many windows properties have been used in this project. Besides the color used in backgrounds, the days of the month are white and the vacations (Sundays) are indicated using the red foreground color.

This simple example would be helpful for beginners as well as intermediate developers to understand some of the basic concepts, like, declaring arrays, using functions, looping, using goto statement, printing output file, handling key press, etc

* The calendar application presented here is a very simple console application developed using c programming language.
* It is built without using graphics properties;instead,it utilizes many windows properties to give the application a colorful look and feel.
* It is compiled in Code::Blocks using GCC compiler.

**Features:**

Windows properties have been used to make the overall application colorful.background color is blue,the days of the month are black and vacations are black in color.

File handling has been used in this application.

It provides a very simple interface and display days,dates months and years based on the input given by the user.

**Source Code:**

#include<stdio.h>

int getfyear(int year)

{

int day=(((year - 1) \* 365) + ((year - 1) / 4) - ((year - 1) / 100) + ((year) / 400) + 1) % 7;

return day;

}

int main()

{

char \*months[]={"January","February","March","April","May","June","July","August","September","October","November","December"};

int daysInMonth[]= {31,28,31,30,31,30,31,31,30,31,30,31};

int i,j,totalDays,weekday=0,spacecounter=0,year;

printf("enter your favourite year:\n");

scanf("%d",&year);

printf("\n\n\*\*\*\*\*\*\*\*\*\*\*\*welcome to %d\*\*\*\*\*\*\*\*\*\n\n");

if((year%4==0&&year%100)||(year%400==0))

daysInMonth[1]=29;

weekday=getfyear(year);

for(i=0;i<12;i++)

{

printf("\n\n-------------------%s------------------\n",months[i]);

printf("\n sun mon tue wed thu fri sat\n\n");

for(spacecounter=1;spacecounter<=weekday;spacecounter++)

{

printf(" ");

}

totalDays = daysInMonth[i];

for(j=1;j<=totalDays;j++)

{

printf("%6d",j);

weekday++;

if(weekday>6)

{

weekday=0;

printf("\n");

}

}

}

return 0;

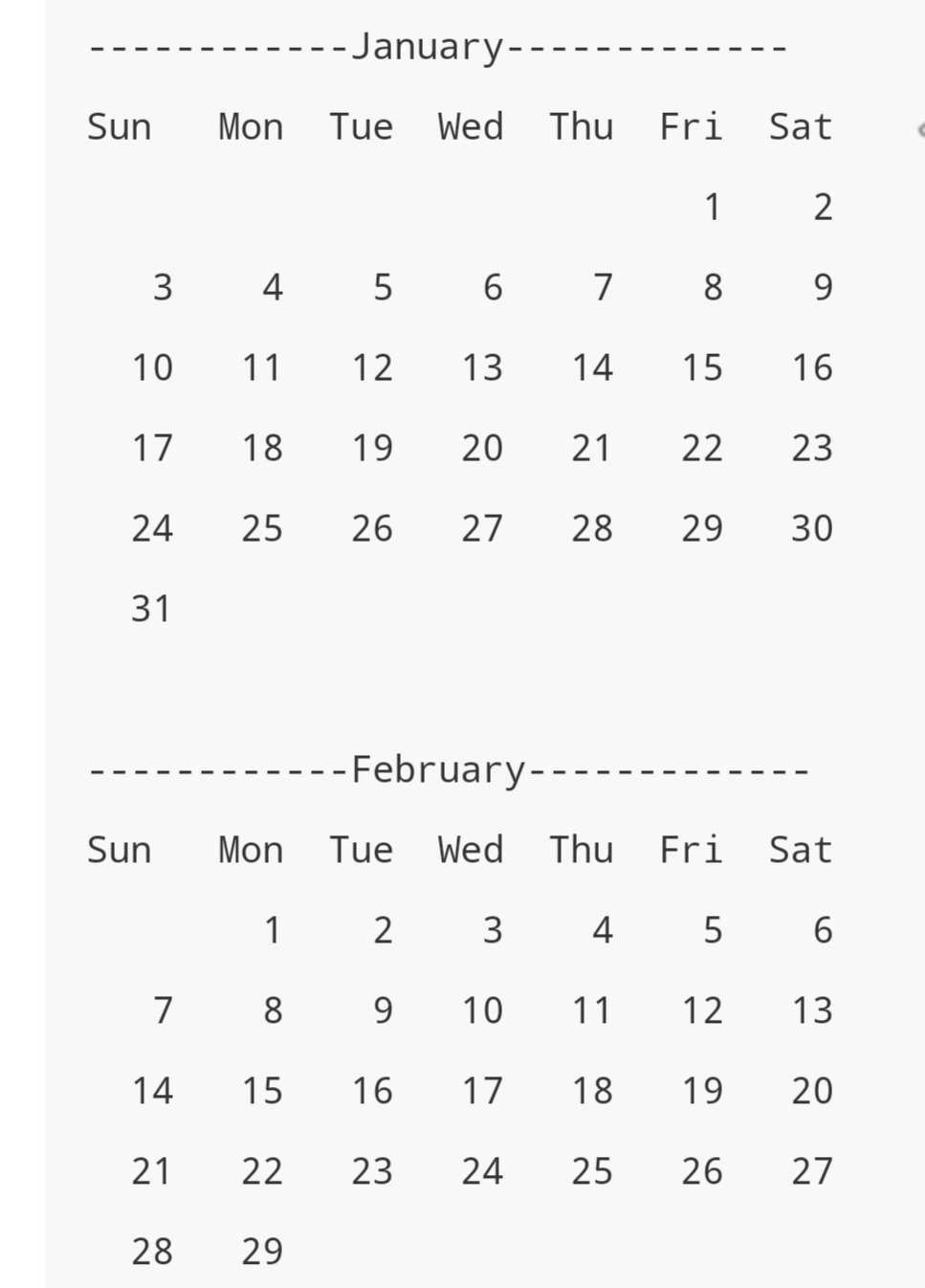
}

**Task:**

* At first , the user needs to provide month and year as input . Example : 2020.This displays the total month of a year .
* The application displays day corresponding to a given date.
* It also displays days and dates corresponding to a particular month and year.
* Months can be navigated using arrow keys .Also ,’n’ and ‘p’ keys can be used to go to next and previous month respectively.

**INPUT :** enter your favourite year 2020

**OUTPUT :**

****

**To be continued December month**

**Code Explanation:**

#include<stdio.h> this is the header file.

Main() , basically code the mian work start main function.

char \*months[]={"January","February","March","April","May","June","July","August","September","October","November","December"};

this is the 12 months name , it,s shows our calendar.

int daysInMonth[]= {31,28,31,30,31,30,31,31,30,31,30,31};

it is total days in a month. Such as January month 31 days,February month 28 or 29 days march month 31 days.

if((year%4==0&&year%100)||(year%400==0))

its return leap year, if this condition satisfied, it return days in month 29 days in February month.

int getfyear(int year)

{

int day=(((year - 1) \* 365) + ((year - 1) / 4) - ((year - 1) / 100) + ((year) / 400) + 1) % 7;

return day;

}

Its return first day of a month , suppose that in 2020 january month start day is on starday,this condition return this day.

First we make two arrays; one with the number of days for a given month and one with all the month names. Note: in both arrays the first position is empty on purpose, we want to use 1 to 12 to keep things simple.

The first function inputyear() is used to get the user input. We ask the user to input a year. Note: that there is no input checking or error handling is performed to keep things simple.)

The next function determinedaycode() is used to get the day number of the first day in that year, so we can print the date on the correct position. (So it is only used for output purposes.)

The next function determineleapyear() is used to determine if input of the user is a leap year. If so, the number if days in February is changed to 29.

The last function calendar() is used to print each month onto the screen. The first [**for loop**](https://www.codingunit.com/c-tutorial-for-loop-while-loop-break-and-continue) is used to loop through all months. We then print the month’s name and all the days of the week. We then use the daycode to position the prompt under the right weekday. Then we print all the dates for one month. The last thing we do is to set the position of the prompt on the right weekday.

That’s all for this C programming tutorial. We hope that you now better understand how to determine days-names or date in a year, month or week and that you can use the calendar example to create your own date/days-names functions.